UNITED STATES PATENT AND TRADEMARK OFFICE

A

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/631,884	08/04/2000	Jyotirmoy Paul	50277-0352	1893

29989 7590 11/02/2005

HICKMAN PALERMO TRUONG & BECKER, LLP
2055 GATEWAY PLACE
SUITE 550
SAN JOSE, CA 95110

EXAMINER	
HILLERY, NATHAN	

ART UNIT	PAPER NUMBER
2176	

DATE MAILED: 11/02/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 09/631,884
Filing Date: August 04, 2000
Appellant(s): PAUL ET AL.

Stoycho D. Draganoff
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 9/14/05 appealing from the Office action mailed 4/14/05.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

A statement identifying the related appeals and interferences which will directly affect or be directly affected by or have a bearing on the decision in the pending appeal is contained in the brief.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

GROUNDS OF REJECTION NOT ON REVIEW

The following grounds of rejection have not been withdrawn by the examiner, but they are not under review on appeal because they have not been presented for review in the appellant's brief. Claims 4 and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over Boag et al. (previously cited) and Bayeh et al. (previously cited),

Monday (previously cited) and Hill et al. (US 6023714 A) as applied to claims 1 – 3, 6 – 10, 12 – 16, 19 – 20, and 23 – 28 above, and further in view of Siyan (previously cited).

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

6,012,098	Bayeh et al.	1-2000
6,589,291	Boag et al.	7-2003
6,480,860	Monday	11-2002
6,023,714	Hill et al.	2-2000

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1 – 3, 6 – 10, 12 – 16, 19 – 20, and 23 – 28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bayeh et al. (previously cited) and further in view of Boag et al. (prev cited), Monday (prev cited) and Hill et al. (US 6023714 A).

Regarding independent claim 1, Bayeh et al. teach that *the role of the data servlet is only to retrieve data from a database* (Column 8, lines 6 – 7) and that *before the data servlet can pass data to another servlet ... it must format that data ... in the preferred embodiment of the present invention, the data servlet formats its output as an XML data stream* (Column 8, lines 13 – 18). Bayeh et al. do not explicitly teach **identifying the particular client device type of the particular client, reading the particular set of metadata that indicates how to convert said XML output to output for said client device type, ... converting the XML output for said client device type, and providing the output for said client device type to said particular client**. However Boag et al. do teach that ... *the selected style sheets are tailored to the client device ... this is done by inspecting the value of the UserAgent field of the HTTP request header with which the document was requested. This UserAgent value will identify the browser running on the client device. (Alternatively, protocols such as CC/PP may be available for querying the device/browser to determine its capabilities dynamically.)* (Column 10, lines 42 – 50), which provides for **identifying the particular client device type of the particular client.** In addition, Boag et al. also teach that *selecting one or more style sheets to transform a particular input document; determining whether a client device is capable of applying the selected style sheets; applying the selected style sheets at the client device when the determining has a positive result; and applying the selected style sheets at a server when the determining has a negative result*

(Column 4, lines 29 - 36) and that *the input document may be encoded in Extensible Markup Language (XML). The style sheets may be encoded in a style sheet language such as Extensible Stylesheet Language (XSL)* (Column 5, lines 8 – 11), which provide for **reading the particular set of metadata based on the particular set of metadata, converting the XML output to output for the particular client device type**. Further, it would be obvious to one with ordinary skill in the art at the time of the invention to know that Boag et al.'s invention is capable of **... converting the XML output to output for the particular client device type**, since Boag et al. further teach that *XML is emerging as a powerful methodology for representing document content, due to its ability to store data in a self-defining, portable manner. Style sheet languages such as XSL, along with their associated processors, are powerful tools for ... transforming documents encoded in one markup language into other markup languages such as HTML (HyperText Markup Language) or WML (Wireless Markup Language)* (Column 2, lines 20 – 28). Also, Boag et al. teach that *if the client device cannot apply style sheets, then they are applied at the server, and the resulting document is sent to the client; otherwise, the document may be sent to the client, where the client will perform the application process* (Abstract, lines 7 – 11), which provides for **providing the output for the particular client device type to the particular client**. It would have been obvious to one with ordinary skill in the art at the time of the invention to combine the invention of Bayeh et al. with that of Boag et al. because such a combination would allow *dynamic determination of the most*

appropriate location for applying style sheets (first sentence of Boag et al.'s Abstract) *used by the rendering servlet for parsing the XML data stream* (last sentence of Bayeh et al.'s Abstract). Neither Boag et al. nor Bayeh et al. explicitly teach **converting the data ... by...** However, Monday teaches that *the bridge identifies and invokes the data access component corresponding to the data request (step 430). In relation to FIG. 2, this means that bridge 125 determines which JavaBean corresponds to the data type of the request. We assume that JavaBean #2 (220) corresponds to the data type of the request. Next, the markup language translator converts the element tags in the markup language data request to corresponding method calls on the applicable data access component (step 440). For the example of FIG. 2, this means that XML translator 226 in bridge 125 converts the request for each piece of data to a corresponding method call on JavaBean #2 (220). Once the element tags in the markup language data request have been translated to method calls, the markup language translator invokes these methods on the data access component, and the retrieved data is placed into a markup language document defined by the DTD corresponding to the data type of the data request (step 450). Thus, XML translator 226 in bridge 125 invokes one or more methods on JavaBean #2 (220), which causes JavaBean #2 (220) to retrieve the requested data from OO database 228. JavaBean #2 (220) then returns the retrieved data to XML translator 226 in bridge 125, which constructs an XML document with the retrieved data. At this point, the return document is sent to the client (step 460),*

which means in FIG. 2 that bridge 125 sends the return document to client 123 via XML interface 224 (Column 7, lines 42 – 67) and that *as new data types are added to the database, corresponding document type definitions (DTDs) may be dynamically generated, allowing a user to access new kinds of data in a database with a software tool that has a user-friendly graphical user interface without having to manually update the software tool for each new data type that is added to the database* (Column 1, lines – 59 – 65), which provide for **converting the data that is to be transmitted from the database application to the particular client into an XML output without regard to the device type of the particular client by: identifying a data type to which the data corresponds, wherein the data type reflects a type of the data that is read out of the database; selecting from a plurality of document type definitions, a document type definition associated with said data type; and converting the data to XML output based on said selected document type definition**. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Boag et al. and Bayeh et al. with that of Monday because such a combination would provide the users of Boag et al. and Bayeh et al. the benefit of *an apparatus and method that defines a markup language for accessing data in a database* (Column 1, lines 49 – 51). Neither Bayeh et al., Boag et al., nor Monday explicitly teach **sets of metadata**. However, Hill et al. do teach that *the document, the layout generator and the style sheets may be created by the author. The author may create a layout*

generator which selects a different style sheet for each type of display. Alternatively, the author may create a layout generator which selects the same style sheet for all display devices with capabilities within a predetermined range. For example, the author may determine that a style sheet entitled "High Resolution" may be used for all display devices with resolutions within a first predetermined range, a style sheet entitled "Medium Resolution" may be used for all display devices with resolutions within a second predetermined range, and a style sheet entitled "Low Resolution" may be used for all other display devices. An authoring tool may assist the author in creating the layout generator and the style sheets. The layout generator may be designed to work with a particular document and a particular set of style sheets or style definitions. Alternatively, the layout generator may be a general purpose layout generator which is designed to work with multiple documents and different sets of style sheets or style definitions (Column 11, lines 4 – 23), which provide for **wherein sets of metadata are each associated with a client device type of a plurality of client device types and indicates how to convert said XML output to output for the client device type; selecting, based on the particular client device type, a particular set of metadata from among the sets of metadata**. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the invention of Hill et al. witht hat of Monday, Boag et al. and Bayeh et al. because such a combination would provide the users of Bayeh et al., Boag et al. and Monday with *a method for dynamically formatting a document based*

upon the capabilities and constraints of a particular output device (Column 2, lines 15 – 17).

Regarding dependent claim 2, Bayeh et al. do not explicitly teach **reading the particular set of metadata includes reading an XSL style sheet ... and converting the output includes applying the XSL style sheet to said XML output.** However, Boag et al. do teach that *selecting one or more style sheets to transform a particular input document; determining whether a client device is capable of applying the selected style sheets; applying the selected style sheets at the client device when the determining has a positive result; and applying the selected style sheets at a server when the determining has a negative result* (Column 4, lines 29 - 36) and that *the input document may be encoded in Extensible Markup Language (XML). The style sheets may be encoded in a style sheet language such as Extensible Stylesheet Language (XSL)* (Column 5, lines 8 – 11), which provide for **reading the particular set of metadata includes reading an XSL style sheet ...** . Further, it would be obvious to one with ordinary skill in the art at the time of the invention to know that Boag et al.'s invention provides for **converting the output includes applying the XSL style sheet to said XML output**, since Boag et al. further teach that *XML is emerging as a powerful methodology for representing document content, due to its ability to store data in a self-defining, portable manner. Style sheet languages such as XSL, along with their associated*

processors, are powerful tools for ... transforming documents encoded in one markup language into other markup languages such as HTML (HyperText Markup Language) or WML (Wireless Markup Language) (Column 2, lines 20 – 28). It would have been obvious to one with ordinary skill in the art at the time of the invention to combine the invention of Bayeh et al. with that of Boag et al. because such a combination would allow *dynamic determination of the most appropriate location for applying style sheets* (first sentence of Boag et al.'s Abstract) *used by the rendering servlet for parsing the XML data stream* (last sentence of Bayeh et al.'s Abstract).

Regarding dependent claim 3, the claim incorporates substantially similar subject matter as claim 1, and is rejected along the same rationale.

Regarding dependent claim 6, Bayeh et al. teach that *in the preferred embodiment of the present invention, the data servlet formats its output as an Extensible Markup Language ("XML") data stream* (Column 8, lines 17 – 19) and that *according to the present invention, the XML data stream 97 is passed on to a "rendering servlet" 85. The function of the rendering servlet 85 is to render the data it receives into a presentation format* (Column 8, lines 30 – 35), which provide for **the XML output includes display instruction data indicating that said data is to be displayed in a first manner.**

Regarding dependent claim 7, Bayeh et al. do not explicitly teach **the step of converting the XML output includes the step of generating output for said particular client device type that causes said data to be displayed in a second manner that is different than said first manner when said particular client device type is not able to display said data in the first manner**. However, Boag et al. do teach that *the style sheet may contain dynamic parameter syntax for an element such as "<HEIGHT>", so that a scaling factor can be applied during the rendering process to scale the document for the height of the particular display on which it will be presented. The translation process will substitute the retrieved value for the dynamic parameter syntax* (Column 12, lines 14 – 20), which provide for **the step of converting the XML output includes the step of generating output for said particular client device type that causes said data to be displayed in a second manner that is different than said first manner when said particular client device type is not able to display said data in the first manner.** It would have been obvious to one with ordinary skill in the art at the time of the invention to combine the invention of Bayeh et al. with that of Boag et al. because such a combination would allow *dynamic determination of the most appropriate location for applying style sheets* (first sentence of Boag et al.'s Abstract) *used by the rendering servlet for parsing the XML data stream* (last sentence of Bayeh et al.'s Abstract).

Regarding independent claim 8, the claim incorporates substantially similar subject matter as claim 1, and is rejected along the same rationale.

Regarding dependent claim 9, the claim incorporates substantially similar subject matter as claim 2, and is rejected along the same rationale.

Regarding dependent claim 10, Bayeh et al. do not explicitly teach **the step of sending the second data to the client includes sending the data to a server to which the client is connected through a wireless connection, and then sending the data from the server to the client over said wireless connection**. However, Boag et al., in Figure 2, teach the capability of **the step of sending the second data to the client includes sending the data to a server to which the client is connected through a wireless connection, and then sending the data from the server to the client over said wireless connection.** It would have been obvious to one with ordinary skill in the art at the time of the invention to combine the invention of Bayeh et al. with that of Boag et al. because such a combination would allow *dynamic determination of the most appropriate location for applying style sheets* (first sentence of Boag et al.'s Abstract) *used by the rendering servlet for parsing the XML data stream* (last sentence of Bayeh et al.'s Abstract).

Regarding independent claim 12, Bayeh et al. teach, in Figure 4, **a database system** (88') **and a database application operatively coupled to**

said database system (82'). . Bayeh et al. also teach that *the role of the data servlet is only to retrieve data from a database 88': it does no presentation formatting of that retrieved data. The data servlet 83 receives the search request 80', queries a database 88' using database query statements 86' appropriate to the particular database, and receives the query results 90'. At that point, the data retrieval function of the data servlet 83 is complete. Before the data servlet 83 can pass data to another servlet for further processing, it must format that data in a manner that allows the next servlet to read and correctly interpret the data. In the preferred embodiment of the present invention, the data servlet formats its output as an Extensible Markup Language ("XML") data stream* (Column 8, lines 6 – 18), which provides for **said database application including application logic ... and an XML processor ...** . Bayeh et al. do not explicitly provide for **an XSL processor ...** . However, it would be obvious to one with ordinary skill in the art at the time of the invention to know that Boag et al.'s invention provides for **an XSL processor...**, since Boag et al. further teach that *XML is emerging as a powerful methodology for representing document content, due to its ability to store data in a self-defining, portable manner. Style sheet languages such as XSL, along with their associated processors, are powerful tools for ... transforming documents encoded in one markup language into other markup languages such as HTML (HyperText Markup Language) or WML (Wireless Markup Language)* (Column 2, lines 20 – 28). It would have been obvious to one with ordinary skill in the art at the time of the invention to combine the invention

of Bayeh et al. with that of Boag et al. because such a combination would allow *dynamic determination of the most appropriate location for applying style sheets* (first sentence of Boag et al.'s Abstract) *used by the rendering servlet for parsing the XML data stream* (last sentence of Bayeh et al.'s Abstract).

Regarding dependent claim 13, Bayeh et al. do not explicitly teach **a plurality of servers operatively coupled to said database application, said plurality of servers including at least a first server ... , a plurality of clients including a first client that interacts with said database application ... .** However, Boag et al. teach that *FIG. 2 illustrates a data processing network 40 in which the present invention may be practiced. The data processing network 40 may include a plurality of individual networks, such as wireless network 42 and network 44, each of which may include a plurality of individual workstations 10. Additionally, as those skilled in the art will appreciate, one or more LANs may be included (not shown), where a LAN may comprise a plurality of intelligent workstations coupled to a host processor. Still referring to FIG. 2, the networks 42 and 44 may also include mainframe computers or servers, such as a gateway computer 46 or application server 47 (which may access a data repository 48). A gateway computer 46 serves as a point of entry into each network 44. The gateway 46 may be preferably coupled to another network 42 by means of a communications link 50a. The gateway 46 may also be directly coupled to one or more workstations 10 using a communications link 50b, 50c* (Column 5, lines 63

– 67 and Column 6, lines 5 – 13), which provides for **a plurality of servers operatively coupled to said database application, said plurality of servers including at least a first server ... , a plurality of clients including a first client that interacts with said database application ...** . It would have been obvious to one with ordinary skill in the art at the time of the invention to combine the invention of Bayeh et al. with that of Boag et al. because such a combination would allow *dynamic determination of the most appropriate location for applying style sheets* (first sentence of Boag et al.'s Abstract) *used by the rendering servlet for parsing the XML data stream* (last sentence of Bayeh et al.'s Abstract).

Regarding independent claim 14, the claim incorporates substantially similar subject matter as claim 1, and is rejected along the same rationale.

Regarding dependent claim 15, the claim incorporates substantially similar subject matter as claim 2, and is rejected along the same rationale.

Regarding dependent claim 16, the claim incorporates substantially similar subject matter as claim 3, and is rejected along the same rationale.

Regarding dependent claim 19, the claim incorporates substantially similar subject matter as claim 6, and is rejected along the same rationale.

Regarding dependent claim 20, the claim incorporates substantially similar subject matter as claim 7, and is rejected along the same rationale.

Regarding dependent claim 23, neither Boag et al. nor Bayeh et al. explicitly teach ... **dumb terminal** ... However, Monday do teach that *terminal interface 140 is used to directly connect one or more terminals 165 to computer system 100. These terminals 165, which may be non-intelligent (i.e., dumb) terminals or fully programmable workstations, are used to allow system administrators and users to communicate with computer system 100* (Column 6, lines 21 – 26), which provides that **the particular client device type indicates at least one of a dumb terminal, a telnet terminal, a bar code scanner and a browser-less device**. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Boag et al. and Bayeh et al. with that of Monday because such a combination would provide the users of Boag et al. and Bayeh et al. the benefit of *an apparatus and method that defines a markup language for accessing data in a database* (Column 1, lines 49 – 51).

Regarding dependent claim 24, the claim incorporates substantially similar subject matter as claim 23, and is rejected along the same rationale.

Regarding dependent claim 25, the claim incorporates substantially similar subject matter as claim 23, and is rejected along the same rationale.

Regarding dependent claims 26 – 28, neither Boag et al. nor Bayeh et al. explicitly teach **the data type indicates...** However, Monday teaches that *the*

query element allows accessing one or more pieces of data based on criteria specified in a database query. A sample data request that conforms to the XML DTD in FIG. 5 is shown in FIG. 6. Line 610 indicates the version of XML, and whether the DTD is a standalone file, which means that the XML file contains all needed data. In line 610, standalone="no" because data is being retrieved from a database into the XML document. Line 620 specifies what type of document will be returned, namely CURRENCYREQUEST for the example of FIG. 6, and also specifies the DTD to be used, namely dataAccess.dtd of FIG. 5. Lines 630 and 660 are the begin and end tags, respectively, for the currency request (Column 8, lines 54 – 66), which provides that **the data type indicates at least one of a data entry form, a queried data, a message, a form level query data and a field level query data**. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Boag et al. and Bayeh et al. with that of Monday because such a combination would provide the users of Boag et al. and Bayeh et al. the benefit of *an apparatus and method that defines a markup language for accessing data in a database* (Column 1, lines 49 – 51).

(10) Response to Argument

Appellant's arguments filed 9/19/05 have been fully considered but they are not persuasive.

Regarding the arguments on pages 7 – 12 in reference to the combination of Bayeh with Boag would change the principle of operation of Bayeh and would render

Bayeh unsatisfactory for its intended purpose, it is the Office's opinion that Appellant is arguing bodily incorporation of references. Appellant is respectively reminded that the test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference; nor is it that the claimed invention must be expressly suggested in any one or all of the references. Rather, the test is what the combined teachings of the references would have suggested to those of ordinary skill in the art. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981). Furthermore, it should be noted that the specific sections or passages of Bayeh cited by the Appellant is not used in the rationale of the above rejection(s) of the claims under 35 USC 103(a).

Specifically, the Office relies mostly on the data servlet of Bayeh. The Office asserts that Bayeh teaches that *the role of the data servlet is only to retrieve data from a database* (Column 8, lines 6 – 7) and that *before the data servlet can pass data to another servlet ... it must format that data ... in the preferred embodiment of the present invention, the data servlet formats its output as an XML data stream* (Column 8, lines 13 – 18). The Office also contends that Bayeh et al. teach that *in the preferred embodiment of the present invention, the data servlet formats its output as an Extensible Markup Language ("XML") data stream* (Column 8, lines 17 – 19) and that *according to the present invention, the XML data stream 97 is passed on to a "rendering servlet" 85. The function of the rendering servlet 85 is to render the data it receives into a presentation format* (Column 8, lines 30 – 35).

Although Bayeh expressly teaches that in the preferred embodiment the rendering servlet must parse the XML data stream and reformat it in HTML as cited by Appellant, the Office does not rely on these specific teachings of Bayeh. Bayeh teaches the general functionality of a rendering servlet, which can be used differently in other embodiments of Bayeh's invention and/or in combination with other teachings such as those of Boag of which one of ordinary skill in the art at the time of the invention would be well aware. The Office is not solely limited to the preferred embodiment of the prior art and does not rely on those teachings in the 103(a) rejection(s) outlined above.

In this case, the Office relies on Bayeh to disclose a data servlet, which retrieves data from a database (Column 8, lines 6 – 7) and formats that data as an XML data stream (Column 8, lines 13 – 18) and Boag to teach dynamic determination of the most appropriate location for applying style sheets (Abstract). Additionally, Boag provides proper motivation to combine its teaching with the disclosure of Bayeh, i.e. to allow dynamic determination of the most appropriate location for applying style sheets. That is to say, the Office does not rely on Bayeh to disclose applying the stylesheet to make the 103(a) rejection for the claims above. The application of the stylsheet taught by Boag can be used as a "rendering servlet" on the XML data stream taught by Bayeh as would have been well-known to one of ordinary skill in the art at the time of the invention.

Regarding the arguments on pages 12 – 15 in reference to the failure of Bayeh, Boag, Monday and Hill in combination or alone to teach, disclose, or suggest the limitations of claim 8, it should be noted that using the broadest, most reasonable interpretation, the Office contends that claim 8 incorporates substantially similar subject

matter as claim 1 and is rejected along the same rationale. Specifically, the Appellant argues that **a plurality of mark-up languages are each associated with one or more client device types of a plurality of client device types** and **selecting based on a client device type to which the output is to be sent, a second mark-up language of said plurality of markup languages that is different than said first mark-up language** are not taught.

However, the Office has interpreted such limitations to be substantially equivalent to the teachings of Boag that *XML is emerging as a powerful methodology for representing document content, due to its ability to store data in a self-defining, portable manner. Style sheet languages such as XSL, along with their associated processors, are powerful tools for ... transforming documents encoded in one markup language into other markup languages such as HTML (HyperText Markup Language) or WML (Wireless Markup Language)* (Column 2, lines 20 – 28). Also, Boag et al. teach that *if the client device cannot apply style sheets, then they are applied at the server, and the resulting document is sent to the client; otherwise, the document may be sent to the client, where the client will perform the application process* (Abstract, lines 7 – 11) as outlined in the rejection of claim 1 under 35 USC 103(a) above.

Specifically, the Offcie asserts that **a plurality of mark-up languages** (HTML, WML, etc) **are each associated with one or more client device types** (wireless device – WML [*Wireless Markup Language*]) **of a plurality of client device types** and **selecting based on a client device type to which the output is to be sent** (*if the client device cannot apply style sheets, then they are applied at the server, and the*

resulting document is sent to the client), **a second mark-up language** (HTML, WML, etc) **of said plurality of markup languages that is different than said first mark-up language** (XML) are taught by Boag. The teachings of Boag were used in the rejection of claim 1; claim 8 surely incorporates substantially similar subject matter as claim 1 and stands rejected along the same rationale.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

NH

25 October 2005

Conferees:

STEPHEN HONG
SUPERVISORY PATENT EXAMINER

Heather Herndon, Supervisory Patent Examiner, GAU 2176

William Bashore, Primary Patent Examiner, GAU 2176

WILLIAM BASHORE
PRIMARY EXAMINER
10/27/2005